On Duplicate Results in a Search Session

Jiepu Jiang, Daqing He, Shuguang Han
School of Information Sciences,
University of Pittsburgh
jiepu.jiang@gmail.com, dah44@pitt.edu, shh69@pitt.edu

ABSTRACT

In this paper, we introduce the PITT group's methods and findings in TREC 2012 session track. After analyzing the search logs in session track 2011 and 2012 datasets, we find that users' reformulated queries are very different from their previous ones, probably indicating their expectations to find not only relevant but also novel results. However, as indicated from our results, a major approach adopted by the session track participants, i.e. using relevance feedback information extracted from previous queries for search, will sacrifice the novelty of results for improving ad hoc search performance (e.g. nDCG@10). Such issues were not disclosed in previous years' session tracks because TREC did not consider the effects of duplicate results in evaluation. Therefore, we proposed a method to properly penalize the duplicate results in ranking by simulating users' browsing behaviors in a search session. A duplicate result in current search will be penalized to a greater extent if it was ranked in higher positions in previous searches or it was returned by more previous queries. The method can effectively improve the novelty of search results and lead to only slight and insignificant drop in ad hoc search performance.

Keywords

Search session; novelty; duplication; query reformulation.

1. DUPLICATE RESULTS IN A SEARCH SESSION: WHERE LIES THE ISSUE?

In TREC 2010 – 2012, the goal of the session track was to investigate whether search performance of the current query in a search session can be improved by using previous user interaction data in the session, including: previous search queries, results, and click through data. The primary evaluation metric adopted by the track guidelines[1][2][3] and overview papers [8–10] is nDCG@10 of the systems' results for the current queries.

In a multi-query search session, one document can be returned in the results of many queries. However, there were debates on whether the duplicate results should be removed. Table 1 shows an example of three different systems' results for q_2 subsequent to the same results for q_1. We assume $D_1 - D_4$ have the same level of relevance. Among the three systems' results for q_2, one can easily agree that S1's are almost useless (only returning duplicate results) and S2's are beneficial (returning a new relevant result D_3 as well as all those found by q_1). However, it is difficult to come to an agreement on whether S3's results are bound to be more/less beneficial than S2's. Compared with S2, S3 returned more new relevant results but less total relevant ones.

[1] http://ir.cis.udel.edu/sessions/guidelines10.html

[2] http://ir.cis.udel.edu/sessions/guidelines11.html

[3] http://ir.cis.udel.edu/sessions/guidelines12.html

Table 1. Examples of duplicate results in a search session.

Reformulation: $q_1 \rightarrow q_2$	q_1's results (S1, S2, S3)	q_2's results		
		S1	S2	S3
Relevant documents returned	D_1 D_2	~~D_1~~ ~~D_2~~	~~D_1~~ ~~D_2~~ D_3	D_3 D_4

Järvelin et al. [6] maintained that duplicate results should not be removed in search because users may overlook relevant results and thus the duplicate ones may still be informative. Therefore, they [6] did not penalize duplicate results at all in evaluation. Kanoulas et al. [7] also argued that removing duplicate results in search may "lead to systems that are less transparent to their users". In evaluation, Kanoulas et al. [7] simply removed the duplicate results from the result list and pushed the subsequent ones up to higher positions, so that they can neither penalize nor take into account the duplicates ones.

However, there are at least three reasons supporting removal of the duplicate results or penalization of their rankings in search:

(1) We find that users' reformulated queries are usually very different from the previous queries in the same session, indicating that finding new relevant results may be partly the expectation of the users for query reformulation. We extracted 128 and 101 query reformulation pairs from the search session logs of the 2011 and 2012 datasets (excluding the current query of each session), respectively. For each query reformulation pair, we calculated the change of search performance (measured by nDCG@10) and the similarity of results (measured by the Jaccard similarity for the pair of queries' top 10 results). As shown in Table 2, on average, we did not find significant change of nDCG@10 on users' reformulated queries, although the sets of results retrieved did change a lot, with relatively low Jaccard similarity with the results of the previous queries.

Table 3 further shows the changes of nDCG@10 and results' similarities for sessions of different task types in 2012 dataset (the task types are manually classified by the Rutgers team [14]). The finding seems consistent among sessions of different task types.

Table 2. Changes of nDCG@10 and results' similarities for query reformulation pairs in TREC 2011 and 2012.

Reformulation: $q_1 \rightarrow q_2$		TREC 2011 (128 query pairs)		TREC 2012 (101 query pairs)	
		mean	SD	mean	SD
nDCG@10 all subtopics	q_1	0.363	0.26	0.227	0.22
	q_2	0.337	0.25	0.195	0.22
	$q_2 - q_1$	-0.026	0.26	-0.031	0.20
	$P(q_1 \neq q_2)$	0.254		0.121	
nDCG@10 current only	q_1	0.147	0.18	-	
	q_2	0.133	0.16		
	$q_2 - q_1$	-0.015	0.18		
	$P(q_1 \neq q_2)$	0.355			
Jaccard(q_1, q_2)		0.109	0.23	0.162	0.21

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
NOV 2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
On Duplicate Results in a Search Session	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
University of Pittsburgh,School of Information Sciences,135 North Bellefield Avenue,Pittsburgh,PA,15260	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

Presented at the Twenty-First Text REtrieval Conference (TREC 2012) held in Gaithersburg, Maryland, November 6-9, 2012. The conference was co-sponsored by the National Institute of Standards and Technology (NIST) the Defense Advanced Research Projects Agency (DARPA) and the Advanced Research and Development Activity (ARDA). U.S. Government or Federal Rights License

14. ABSTRACT

In this paper, we introduce the PITT group?s methods and findings in TREC 2012 session track. After analyzing the search logs in session track 2011 and 2012 datasets, we find that users? reformulated queries are very different from their previous ones, probably indicating their expectations to find not only relevant but also novel results. However, as indicated from our results, a major approach adopted by the session track participants, i.e. using relevance feedback information extracted from previous queries for search, will sacrifice the novelty of results for improving ad hoc search performance (e.g. nDCG@10). Such issues were not disclosed in previous years? session tracks because TREC did not consider the effects of duplicate results in evaluation. Therefore, we proposed a method to properly penalize the duplicate results in ranking by simulating users? browsing behaviors in a search session. A duplicate result in current search will be penalized to a greater extent if it was ranked in higher positions in previous searches or it was returned by more previous queries. The method can effectively improve the novelty of search results and lead to only slight and insignificant drop in ad hoc search performance.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**5**	

Table 3. Changes of nDCG@10 and results' similarities for query reformulation pairs in sessions of different task types.

	Know-Subject N = 13		Know-Item N = 46		Exploratory N = 32		Interpretive N = 10	
	mean	SD	mean	SD	mean	SD	mean	SD
q_1	0.131	0.20	0.204	0.23	0.296	0.21	0.235	0.24
q_2	0.100	0.18	0.180	0.23	0.259	0.22	0.186	0.23
$q_2 - q_1$	-0.031	0.26	-0.024	0.15	-0.037	0.21	-0.049	0.31
$P(q_1 \neq q_2)$	0.681		0.295		0.319		0.632	
Jaccard	0.142	0.22	0.141	0.18	0.209	0.24	0.135	0.22

(2) We noticed that one major approach being adopted by the participants in session track [1, 4, 5, 11], i.e. using previous search queries as relevance feedback information, may make the search results of the current query more similar to the results of previous searches. Therefore, although the approach improved nDCG@10, it is unclear whether the improvements come from returning new relevant results or the duplicate ones found in previous searches.

Figure 1 shows the average Jaccard similarity between the current query's results and each of the previous query's results for our run "PITTSHQM", which used the mentioned approach in RL2–4. It is indicated that the results of RL2–4 are more similar to previous queries' results than those of RL1 (in which only the current queries were used for search). Note that the seemingly low Jaccard similarity values in Figure 1 may be underestimated due to the difference between our system and the system used for collecting search logs.

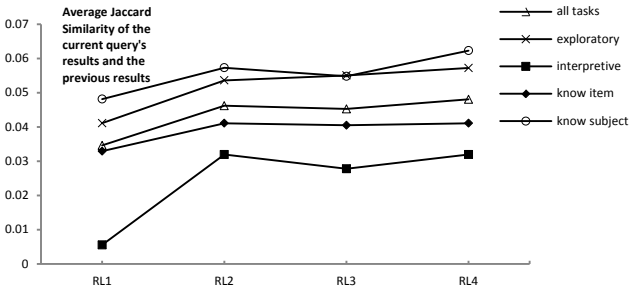

Figure 1. Average Jaccard similarity between the current query's results and previous results for run "PITTSHQM".

(3) Even though sometimes users may not want the duplicate results to be removed or penalized, it does not constitute a reason for against removing or penalizing the duplicate results. In fact, in a real system, we can provide both results with and without removing the duplicates and let the users decide which one to use. Moreover, we also believe that the question whether we should remove the duplicate results can be modeled based on users' previous behaviors.

Therefore, in TREC session track 2012, we focus on how to develop appropriate methods to penalize the rankings of duplicate results based on users' previous behaviors in the session. The rest of the paper is organized as follows: section 2 introduces our methods for TREC 2012; section 3 introduces the experiment settings; section 4 evaluates our results and draws conclusions.

2. METHODS

We use a language modeling approach for retrieval. A document d will be ranked by $P(d|q, s)$: q is the current query for search in the ongoing session; s is the user's past search behaviors in the session. As in Eq(1), applying Bayes' theorem, we can equivalently rank documents by the product of $P(q|d, s)$ and $P(d|s)$. We further model $P(q|d, s)$ as d's topical relevance to the query q in the session context s, and $P(d|s)$ as the novelty of d to the user when browsing the current query's results.

$$P(d \mid q,s) \propto P(q \mid d,s) \cdot P(d \mid s) \tag{1}$$

2.1 Topical Relevance

Literally, $P(q|d, s)$ suggests a query generation process that q is generated from not only the document d but also the session context s. We can also explain $P(q|d, s)$ as the likelihood that the user issues a query q in the specific session context s for retrieving the document d. Apparently, this suggests an extension to the query likelihood language model (LM) framework [18, 20].

Similarly, we can give out an extension to the KL-Divergence LM framework [12, 21] in multi-query search session. As in Eq(2), $P(q|d, s)$ is proportional to $P(q, s|d, s)$. Thus, we can estimate two language models $\theta_{q,s}$ and $\theta_{d,s}$, the session contextual query model and document model, and rank documents by the KL-Divergence between $\theta_{q,s}$ and $\theta_{d,s}$. We finally calculate the relevance scores by $\sum_{t \in \theta_{q,s}} P\left(t \mid \theta_{d,s}\right)^{P\left(t \mid \theta_{q,s}\right)}$, which is equivalent to $KLD(\theta_{q,s}||\theta_{d,s})$ in ranking and can be easily implemented using indri query language.

$$\begin{aligned} P(q \mid d,s) &\propto P(q,s \mid d,s) \\ &\overset{rank}{=} \sum_{t \in \theta_{q,s}} P\left(t \mid \theta_{d,s}\right)^{P\left(t \mid \theta_{q,s}\right)} \overset{rank}{=} KLD\left(\theta_{q,s} \,\|\, \theta_{d,s}\right) \end{aligned} \tag{2}$$

Although $\theta_{q,s}$ and $\theta_{d,s}$ provide us with interesting opportunities for modeling, this year we only adopt very simple methods for $\theta_{q,s}$ and $\theta_{d,s}$, so that we can focus on our research question, i.e. how to consider duplicate results in a session. We simply estimate $\theta_{d,s}$ as θ_d, the plain document language model with Dirichlet smoothing [20], as in Eq(3). As in Eq(4), we estimate $\theta_{q,s}$ by interpolating different query models: $P_{MLE}(t|q)$ and $P_{MLE}(t|q_s)$, respectively, are models estimated from the latest query q and the past queries q_s by maximum likelihood estimation (MLE); $P_{fb}(t|\theta_{q,s})$ is a relevance feedback query model.

$$P\left(t \mid \theta_{d,s}\right) \approx \hat{P}\left(t \mid \theta_d\right) = \frac{c(t,d) + \mu \cdot P\left(t \mid C\right)}{\sum_{t_i \in d} c(t_i,d) + \mu} \tag{3}$$

$$\begin{aligned} \hat{P}\left(t \mid \theta_{q,s}\right) = \left(1-\lambda_{fb}\right) \cdot \left\{\left(1-\lambda_{prev}\right) \cdot P_{MLE}\left(t \mid q\right) + \lambda_{prev} \cdot P_{MLE}\left(t \mid q_s\right)\right\} \\ + \lambda_{fb} \cdot P_{fb}\left(t \mid \theta_{q,s}\right) \end{aligned} \tag{4}$$

Specifically, we estimate different query models for RL1–4 runs. RL1 runs only use $P_{ML}(t|q)$. RL2 runs combine $P_{ML}(t|q)$ with $P_{ML}(t|q_s)$. RL3 and RL4 runs interpolate RL2 runs' models with different relevance feedback query models: for RL3 runs, $P_{fb}(t|\theta_{q,s})$ is estimated based on RL2 runs' top ranked results using RM1 relevance model [13, 15]; for RL4 runs, we estimate $P_{fb}(t|\theta_{q,s})$ as the mixture model of all clicked documents' MLE document models (we assign each clicked document the same weight).

Technically, the topical relevance scores are calculated using exactly the same methods we adopted last year [4]. Here we show the methods in [4] suggests an extension to the language modeling methods for ad hoc search [20, 21] in multi-query search session. Similar methods were also adopted by many other TREC session participants [1, 11] and can be at least traced back to Shen et al.'s models in [19] (the FixInt method).

2.2 Browsing Novelty

We model the user's browsing novelty in a multi-query session by $P(d|s)$, which can be explained as: the probability that the user, after several rounds of searches and interactions (s), will still be interested in examining d.

A document may lose its attractiveness for at least two reasons: first, it was examined by the user in past searches; second, other documents examined previously contain the same or very similar information. We focus on the first type of novelty due to the lack of information for studying and evaluating the second type (e.g. mapping between documents and subtopics).

We assume the following models for the user's behaviors prior to the current query q:

M1: The user examines results in a list by sequence. The user will always examine the first result in a list. After examine each result, the user has probability p to continue examining the next one, and probability $1 - p$ to stop (either to reformulate a new query for search or to terminate the current session).

M2: For each time the user examines a result, it has probability β that the result will lose its attractiveness to the user in the rest of the search session.

Here, M1 models user's browsing behaviors in a search session. We adopt the same browsing model used in rank-biased precision (RBP) [17]. A similar model has been adopted in [7] for evaluating a whole search session's performance. However, M1 differs from the model in [7] in that we do not count any probability for the case that the user terminates the session prior to q (as modeled by p_{reform} in [7]). This is because, in a static session dataset such as those in TREC session track, we can only observe the static session data based on the fact that the user had chosen to reformulate queries in M1. Thus, it seems inconsistent for [7] to consider p_{reform} in such datasets. If the user terminated the session prior to q, we will not be able to observe the static session data.

M2 is not an actual "model" on the process that a document loses its attractiveness. But M2 can roughly model the effects that the attractiveness of a document is lost due to many complex user factors in interactive search, for example:

Users' browsing styles and efforts: some users may quickly scan results, while some others may carefully examine one by one. Users of different styles may have different chances of missing important information in a document.

Users' background knowledge and familiarity with the topic: a user's background knowledge and familiarity with the topic may influence whether, after examining a result, the user can understand the major information in the result.

Here we simply set up a value for β intuitively and left the modeling of user factors in β for future works. According to M1 and M2, as in Eq(5), a document d can keep its attractiveness if and only if it did not lose attractiveness in any of the previous searches. In Eq(5): $R^{(i)}$ refers to the results for the ith query in the session (assuming q is the nth query); $P_{\text{examine}}(d|R^{(i)})$ is the probability that d will be examined when the user browses results $R^{(i)}$, as calculated in Eq(6); rank(d, i) is the rank of d in $R^{(i)}$.

$$P(d \mid s) = 1 - \prod_{i=1}^{n-1} \left(1 - \beta \cdot P_{\text{examine}}(d \mid R^{(i)})\right) \quad (5)$$

$$P_{\text{examine}}(d \mid R^{(i)}) = \begin{cases} p^{rank(d,i)-1} & d \in R^{(i)} \\ 0 & d \notin R^{(i)} \end{cases} \quad (6)$$

According to Eq(5) and Eq(6), a duplicate document will be discounted to a greater extent if: the document appeared in more previous queries' results; the document was at higher positions in previous results; a greater value of either p or β is assigned. Let $S\{d_1, d_2, \ldots, d_{10}\}$ be a result list of 10 documents. Figure 2 shows $P(d_i|s)$ for the same 10 documents after the user viewed S once. We used a similar model in [3] for evaluating performance of query reformulations in a search session.

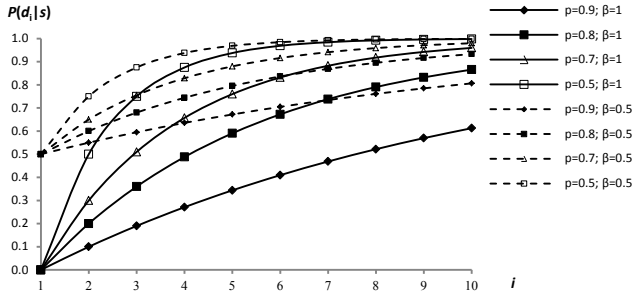

Figure 2. Discounting of results' attractiveness to the user.

3. EXPERIMENTS

We submitted 4 runs, as summarized in Table 4. The parameter settings are summarized in Table 5. We implement Eq(2) using Indri's query language. We build index and search on a subset of Clueweb09b dataset for only those documents that have Waterloo spam rank scores ≥ 70.

PITTSHQM: only considered topical relevance; used unigram language model.

PITTSHQMsdm: only considered topical relevance; used sequential dependence model [16].

PITTSHQMnov: considered both topical relevance and browsing novelty; used unigram language model

PITTSHQMsnov: considered both topical relevance and browsing novelty; used sequential dependence model.

Table 4. Summarization of runs.

Runs/Methods	Topical Relevance	Browsing Novelty	SDM
PITTSHQM	Y	N	N
PITTSHQMsdm	Y	N	Y
PITTSHQMnov	Y	Y	N
PITTSHQMsnov	Y	Y	Y

Table 5. Summarization of parameters.

Related Models	Parameter Settings
Document Model	μ = 3,500
Session History Query Model	λ_{prev} = 0.4
Relevance Feedback Query Model	λ_{fb} = 0.2
	# fb docs = 10
	# fb terms = 20
Sequential Dependence Model	w_{term} = 0.85
	$w_{\#\text{ow2}}$ = 0.09
	$w_{\#\text{uw8}}$ = 0.06
Browsing Novelty	p = 0.8
	β = 0.8
Waterloo Spam Rank Scores	≥ 70

4. EVALUATION

Table 6 shows nDCG@10 for the submitted runs (using all qrels for evaluation without considering duplicate results). We find very similar results to what we found last year: nDCG@10

can be improved substantially by combining the past search queries with the current query, but further applying relevance feedback query models to RL2 runs seems not helpful.

We focus on the effectiveness of the browsing novelty model (PITTSHQM vs. PITTSHQMnov, and PITTSHQMsdm vs. PITTSHQMsnov). As indicated in Table 6, there are slight drops in nDCG@10 (about 2%, not significant) after applying browsing novelty model. Table 7 shows nDCG@10 for the submitted runs using qrels that consider the relevance of duplicate results as zero (we refer to the nDCG@10 using this qrels as **nDCG@10-nov**). As shown in Table 7, after applying the browsing novelty model, nDCG@10-nov improved significantly in most of the cases (by about 8% – 10%). Table 8 shows the average Jaccard similarity between the current query's results and previous queries' results for the submitted runs. After applying the browsing novelty model, the similarity between current query's results and previous queries' results also dropped greatly.

Results in Table 6, 7, and 8 indicate the effectiveness of the browsing novelty model in finding new relevant results. In general, it seems worthwhile to apply the browsing novelty model, as it significantly improved nDCG@10-nov while led to only slight and insignificant drop in nDCG@10.

We further calculate the rank correlation between the top 10 results of PITTSHQM.RL1 and PITTSHQMnov.RL1. Figure 2 shows the results. In 55 out of 98 sessions, the top 10 results' rankings were affected by the browsing novelty model (with average tau = 0.71), but their nDCG@10 did not change much (with only -0.007 change in nDCG@10). After analyzing the results, we find: the browsing model will not only penalize the relevant documents ranked at high positions in previous searches, but also shuffle some new relevant results to higher positions so that the nDCG@10 scores will not be affected much. For example, Table 4 shows the shuffling of search results for session No. 47. A relevant document "clueweb09-enwp01-63-10556" was ranked at the top position by the first query in the session. The document will be discounted to very low positions so that other relevant documents can be shuffled to higher positions.

Table 6. nDCG@10 of the submitted runs.

	TP	Nov	SDM	RL1	RL2	RL3	RL4
PITTSHQM	Y	N	N	0.256	$0.310^{\uparrow}$	$0.322^{\uparrow}$	$0.315^{\uparrow}$
PITTSHQMnov	Y	Y	N	0.252	$0.301^{\uparrow}$	$0.315^{\uparrow}$	$0.307^{\uparrow}$
PITTSHQMsdm	Y	N	Y	0.262	$0.307^{\uparrow}$	$0.310^{\uparrow}$	$0.310^{\uparrow}$
PITTSHQMsnov	Y	Y	Y	0.254	$0.297^{\uparrow}$	$0.301^{\uparrow}$	$0.302^{\uparrow}$

↑: RL2–4's results are significantly better than RL1's (p<0.05) by 2 tail paired t-test.

Table 7. nDCG@10-nov of the submitted runs (shown documents in previous queries of the session are considered duplicates and their relevance are downgraded to zero).

	TP	Nov	SDM	RL1	RL2	RL3	RL4
PITTSHQM	Y	N	N	0.231	$0.275^{\uparrow}$	$0.288^{\uparrow}$	$0.278^{\uparrow}$
PITTSHQMnov	Y	Y	N	0.250^{*}	$0.300^{\uparrow *}$	$0.315^{\uparrow *}$	$0.306^{\uparrow *}$
PITTSHQMsdm	Y	N	Y	0.234	$0.265^{\uparrow}$	$0.270^{\uparrow}$	$0.270^{\uparrow}$
PITTSHQMsnov	Y	Y	Y	0.250	$0.292^{\uparrow *}$	$0.296^{\uparrow *}$	$0.296^{\uparrow *}$

↑: RL2–4's results are significantly better than RL1's (p<0.05) by 2-tail paired t-test;
*: differences between PITTSHQM vs. PITTSHQMnov and PITTSHQMsdm vs. PITTSHQMsnov are significant (p<0.05) by 2-tail paired t-test.

Table 8. Average Jaccard similarity between the current query's results and previous results for submitted run.

	TP	Nov	SDM	RL1	RL2	RL3	RL4
PITTSHQM	Y	N	N	0.035	0.046	0.045	0.048
PITTSHQMnov	Y	Y	N	0.003	0.004	0.002	0.002
PITTSHQMsdm	Y	N	Y	0.030	0.041	0.041	0.041
PITTSHQMsnov	Y	Y	Y	0.004	0.006	0.005	0.006

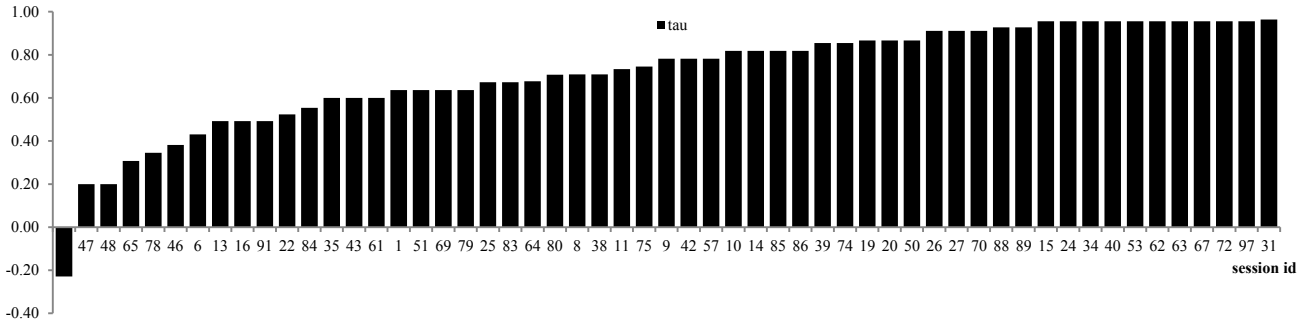

Figure 3. Correlation of top 10 results between PITTSHQM.RL1 and PITTSHQMnov.RL1 (Kendall's tau).

Table 9. Shuffling of results in session #47 after applying browsing novelty model.

q_1 = "pseudocyesis"		q_2 = "pseudocyesis epidemiology"		q = "pseudocyesis history"				
				PITTSHQM.RL1		PITTSHQMnov.RL1		
rank	**docno**	**rank**	**docno**	**rank**	**docno**	**rank change**		**docno**
1	enwp01-63-10556	1	enwp01-23-15772	1	enwp01-63-10556	↓	2→1	enwp00-68-14496
2	en0038-44-08898	2	enwp00-88-14910	2	enwp00-68-14496	↑	3→2	enwp02-13-04273
3	en0013-47-24913	3	en0060-14-21952	3	enwp02-13-04273	↑	4→3	enwp01-83-08322
4	en0121-70-04288	4	en0006-59-10549	4	enwp01-83-08322	↑	8→4	enwp00-86-21481
5	en0047-21-02636	5	en0009-11-14983	5	enwp01-56-06800	↓	10→5	enwp00-94-21656
6	enwp01-80-10554	6	en0011-66-21877	6	enwp01-66-10938	↓	9→6	enwp00-98-19091
7	en0123-83-35172	7	en0074-17-31531	7	enwp01-51-08462	=	7→7	enwp01-51-08462
8	en0063-23-33834	8	en0005-88-05908	8	enwp00-86-21481	↑	5→8	enwp01-56-06800
9	en0065-33-00328	9	en0004-33-02114	9	enwp00-98-19091	↑	6→9	enwp01-66-10938
10	en0092-76-41724	10	en0013-29-10622	10	enwp00-94-21656	↑	12→10	enwp02-21-21481
				…			…	
				12	enwp02-21-21481	↑	1→36	enwp01-63-10556

5. CONCLUSION AND FUTURE WORKS

In TREC 2012 session track, we mainly focus on studying the proposed browsing novelty model. The evaluation results indicate that it is beneficial and worthwhile to apply the browsing novelty model to penalize duplicate results.

It should be noted that the influence of the duplicate results may be underestimated in TREC session track mainly because the participants' systems are usually very different from those used for collecting search logs. Thus, the overlap between the systems' results and previous queries' results should be higher than what are shown in Table 8. Therefore, it is still unclear to what degree the duplicate results can influence search systems in a session and how effective the browsing novelty model can solve the issues.

Besides, it may be problematic to simply consider duplicate results' relevance as zero in the evaluation (i.e. nDCG@10-nov). Here we suggest two alternative evaluation methods:

(1) A model-free approach. Enlightened by the interactive search and judge method for collecting qrels [2], we can ask the users to freely search in an interactive search system, saving each relevant document **if and only if** the user believes the document is relevant and should not be presented again in search results. Using this approach, we can collect the user's whole search history as a static search session (similar to the current search logs in session track), along with the time-sensitive qrels for the session: each relevant result is associated with the time in the session it was recognized by the user as relevant and obsolete. When evaluating a query's search performance, we only use the relevant results saved later than the query as qrels. However, this approach also requires extensive endeavors in developing new datasets.

(2) Using existing datasets and qrels but modeling on novelty in evaluation metrics, such as the *irel*-series metrics we proposed in [3]. However, it is very likely that the evaluation metrics will be biased to the search systems that applied a similar model (for example, the evaluation metrics in [3] will be biased to the search methods proposed in this paper, because they use the same browsing model).

6. REFERENCES

[1] Albakour, M.-D. et al. 2010. University of Essex at the TREC 2010 Session Track. In the 19th Text REtrieval Conference Notebook Proceedings (TREC 2010).

[2] Cormack, G. V et al. 1998. Efficient construction of large test collections. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'98).

[3] Jiang, J. et al. 2012. Contextual evaluation of query reformulations in a search session by user simulation. In Proceedings of the 21st ACM international conference on Information and knowledge management (CIKM'12).

[4] Jiang, J. et al. 2011. Pitt at TREC 2011 session track. In Proceedings of the 20th Text REtrieval Conference, (TREC 2011).

[5] Jiang, J. et al. 2012. PITT at TREC 2012 Session Track: Adaptive Browsing Novelty in a Search Session. In 21st Text REtrieval Conference Notebook Proceedings (TREC 2012).

[6] Järvelin, K. et al. 2008. Discounted Cumulated Gain Based Evaluation of Multiple-Query IR Sessions. In LNCS 4956: Proceedings of the 30th European Conference on Information Retrieval (ECIR'08).

[7] Kanoulas, E. et al. 2011. Evaluating multi-query sessions. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (SIGIR'11).

[8] Kanoulas, E. et al. 2012. Overview of the TREC 2012 Session Track DRAFT Notebook Version - please do not distribute. In 21st Text REtrieval Conference Notebook Proceedings (TREC 2012).

[9] Kanoulas, E. et al. 2011. Session Track 2011 Overview. In 20th Text REtrieval Conference Notebook Proceedings (TREC 2011).

[10] Kanoulas, E. et al. 2010. Session track overview. In 19th Text REtrieval Conference Notebook Proceedings (TREC 2010).

[11] Kharazmi, S. et al. 2010. RMIT University at TREC 2010: Session Track. In 19th Text REtrieval Conference Notebook Proceedings (TREC 2010).

[12] Lafferty, J. and Zhai, C. 2001. Document language models, query models, and risk minimization for information retrieval. In Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'01).

[13] Lavrenko, V. and Croft, W.B. 2001. Relevance based language models. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'01).

[14] Liu, C. et al. 2012. Rutgers at the TREC 2012 Session Track. In 21st Text REtrieval Conference Notebook Proceedings (TREC 2012).

[15] Lv, Y. and Zhai, C. 2009. A comparative study of methods for estimating query language models with pseudo feedback. Proceedings of the 18th ACM conference on Information and knowledge management (CIKM'09).

[16] Metzler, D. and Croft, W.B. 2005. A Markov random field model for term dependencies. Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'05).

[17] Moffat, A. and Zobel, J. 2008. Rank-biased precision for measurement of retrieval effectiveness. ACM Trans. Inf. Syst. 27(1).

[18] Ponte, J.M. and Croft, W.B. 1998. A language modeling approach to information retrieval. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'98).

[19] Shen, X. et al. 2005. Context-sensitive information retrieval using implicit feedback. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'05).

[20] Zhai, C. and Lafferty, J. 2004. A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. 22(2).

[21] Zhai, C. and Lafferty, J. 2001. Model-based feedback in the language modeling approach to information retrieval. In Proceedings of the tenth international conference on Information and knowledge management (SIGIR'01).